

6-DOF AEROBRAKING TRAJECTORY RECONSTRUCTION BY USE OF INERTIAL MEASUREMENT UNIT (IMU) DATA FOR THE IMPROVEMENT OF AEROBRAKING NAVIGATION

Moriba K. Jah and Michael E. Lisano¹

For any interplanetary mission, there are certain types of data that are used as a means of determining both the position and velocity of a spacecraft. NASA's Deep Space Network (DSN) is employed for the purpose of transmitting and receiving data to and from the spacecraft, respectively. For this exchange of information to take place, both the DSN and spacecraft antennae must be pointed towards each other. This mutual geometry is not maintained throughout aerobraking, specifically while the spacecraft is within the atmosphere. Some spacecraft are equipped with an Inertial Measurement Unit (IMU), which typically is comprised of gyroscopes and accelerometers. The IMU provides information about the spacecraft's non-conservative acceleration and angular motion. Since the spacecraft loses telemetry during the drag pass, this research focuses upon the possibility of using IMU data as a means of augmenting the current state knowledge of the spacecraft. This knowledge could also assist in obtaining best estimates for subsequent periapse times and altitudes (integral aspect of aerobraking operations). This research focuses upon the use of the IMU data (collected during the radiometric data gap) as a means of augmenting aerobraking navigation capabilities.

1. INTRODUCTION

For any interplanetary mission, there are certain types of data that are used as a means of determining both the position and velocity of a spacecraft. The data types currently in use are Doppler, Range, Optical, and Δ VLBI. All of them are radiometric with the exception of the Optical data type. NASA's Deep Space Network (DSN) is employed for the purpose of transmitting and receiving data to and from the spacecraft, respectively. For this exchange of information to take place, both the DSN and spacecraft antennae must be pointed towards each other.

To decrease propulsive expenses for a given mission, a spacecraft may be initially placed into a highly eccentric orbit, with periapse located within a planet's atmospheric influence. Outside this atmospheric influence, this highly eccentric orbit would theoretically remain unchanged in size due to the spacecraft's presence in a conservative field (gravitational). However, because of the viscous effects that the spacecraft

¹ Members Technical Staff, Navigation and Mission Design section, Jet Propulsion Laboratory, California Institute Of Technology
Pasadena, CA 91109

experiences within the atmosphere, the presence of a non-conservative force is encountered, commonly known as “drag”. Each “pass” through the atmosphere (drag pass), the spacecraft’s orbit will decrease in size and with it, its period. This decrease is allowed to take place until such time at which operations deems apoapse to have reached its desired altitude. Then, the spacecraft performs a periapse-raising maneuver and the orbit is essentially circularized or is placed in its final configuration. This method of orbit size reduction is commonly known as “aerobraking”. Aerobraking can be viewed as a means to achieve a change in velocity provided *gratis* by the atmosphere, which would otherwise have to be provided by the spacecraft thrusters. The following figure illustrates this process.

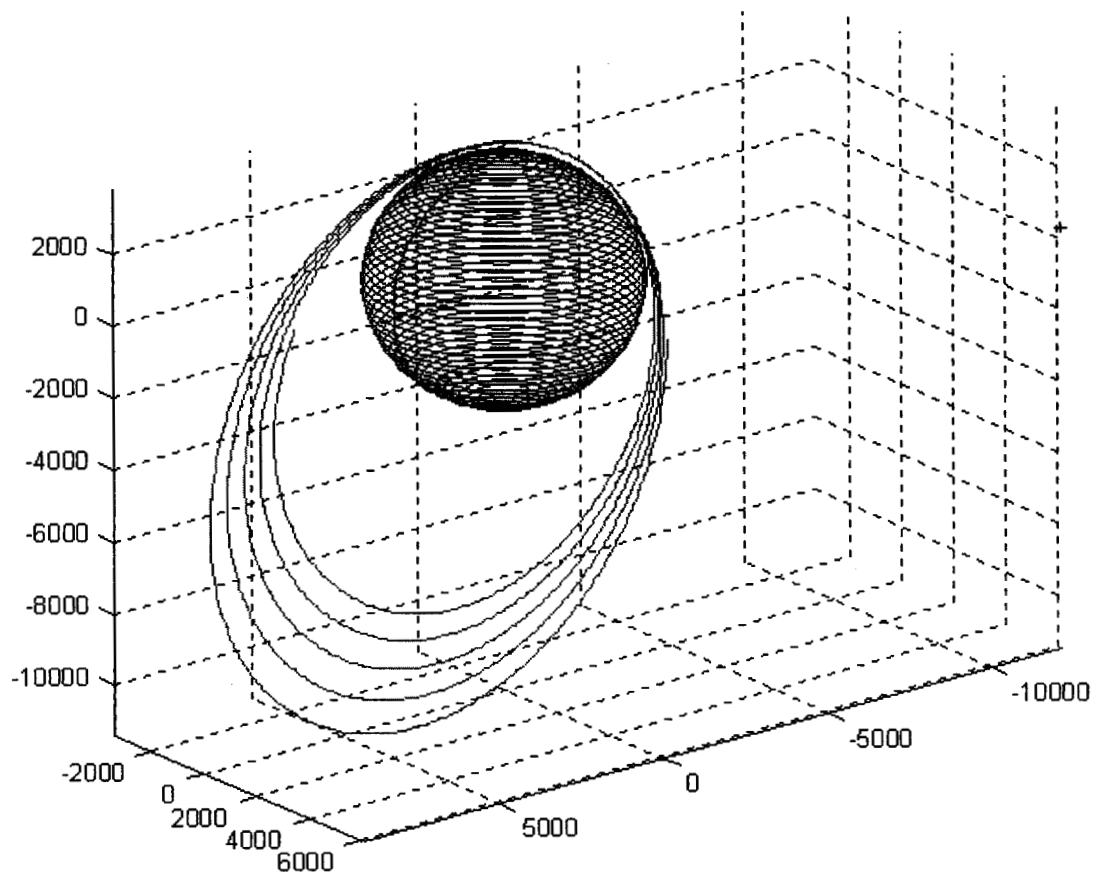


Figure 1. Example of Several Aerobraking Orbits

From an operations perspective, aerobraking navigation is performed as follows:

1. Initial Conditions (IC) are obtained from the previous drag pass.
2. Radiometric data is fit up to the current drag-pass attitude slew.
3. Small forces telemetry is collected during the drag pass and “dumped” to Earth in telemetry.
4. Radiometric data is collected and fit, along with the small forces data, in order to reconstruct the drag pass.

5. Subsequent orbits are predicted with MarsGRAM
6. Predictions are compared to a nominal baseline and deviations are analyzed for aerobraking corridor control strategy implementations.
7. Navigation products (namely trajectories, salient information, and maneuvers if required) are delivered to parties of interest.

For the sake of clarity, the following figure illustrates this process:

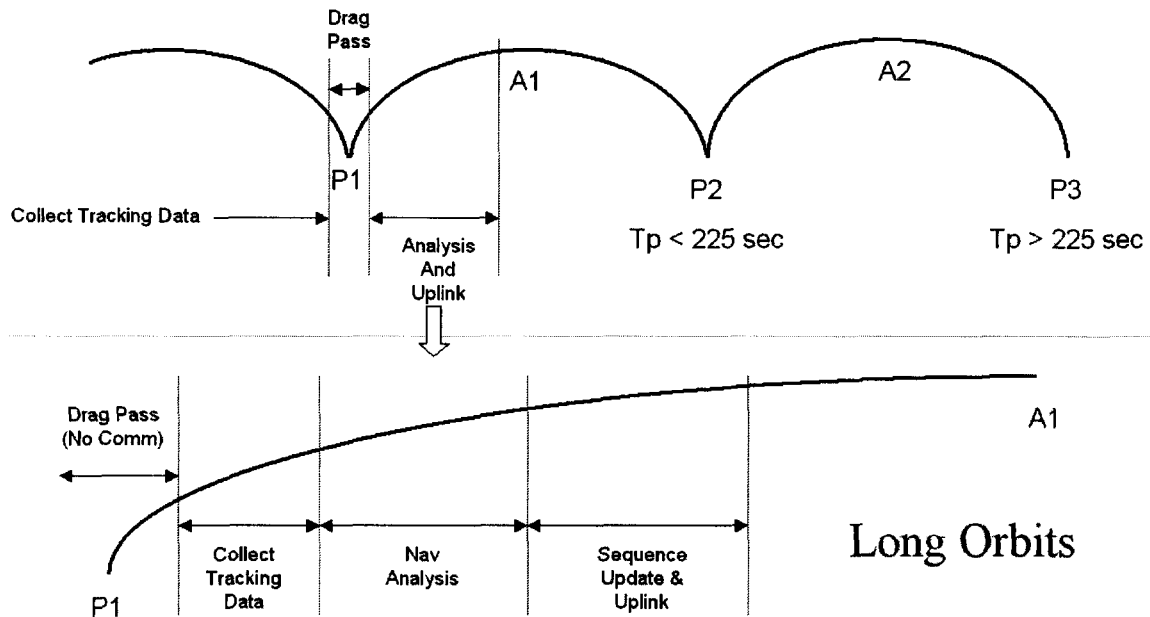


Figure 2. Long Orbit Period Aerobraking Navigation Strategy.

The navigation team is required to be able to predict future periapse times (T_p) to within 225 seconds. This timing uncertainty is dominated by the assumption made on future drag-pass atmospheric uncertainty. Currently, this assumption is 35% 1-sigma, orbit-to-orbit. To first order, the expected orbit period change per drag-pass will indicate how well future periapses can be predicted. This simplifying assumption is supported by covariance studies. For example, if the next orbit's period change is 1000 seconds, then the uncertainty will be 1050 seconds (105% 3-sigma). If the period could change by 1050 seconds, then the periapse time can be off by this amount. This is beyond the 225 second requirement. For shorter orbit periods where the change in period is on the order of 30 seconds, navigation can predict on the order of 7 orbits before violating the 225 second constraint. However, the navigation team usually has 3 hours from the time of last periapse passage to reconstruct the drag-pass and deliver the appropriate navigation products. For shorter orbits, a couple of orbits have occurred by the time a sequence is built and uplinked to the spacecraft. Again, since the period change is smaller, this can be tolerated.

Aerobraking, a tried-and-true mission propellant-saving technique for planetary orbiters, has operational caveats:

1. The spacecraft must slew into the aerobraking orientation prior to each drag pass. There is a loss of radiometric tracking precisely when the spacecraft “flies” through the most dynamically changing and unknown portion of its trajectory. This leads to a significant increase in the spacecraft’s post-pass state (position and velocity) uncertainty.
2. When reconstructing the drag pass, it is assumed that the spacecraft’s total change in velocity due to the atmospheric effects is purely due to drag. In practice, the aerobraking orbits do not tend to fit the radiometric data unless residual noise is modeled as artificial dynamic acceleration events; it is likely that the residual noise is due to the lack of modeling of aerodynamic lift and side-force.
3. The aerobraking orbit reconstruction process is very time consuming (i.e. lasting several hours for each orbit) and workforce intensive (9 navigators for the Mars Odyssey aerobraking operations).
4. All of the spacecraft events occur on a ground-generated timeline (i.e. a sequence of commands). At times, up to 3 sequences must be generated and successfully uplinked to the spacecraft every 24 hours. The personnel required to perform this task constitutes an additional operational cost.
5. Spacecraft events take place at times relative to the predicted time of periapse. Any error in this prediction (> 225 seconds) could lead to:
 - a. aerobraking corridor control maneuver errors and thus inefficient propellant usage.
 - b. aerobraking drag pass attitude configuration slewing at off-nominal times, capable of inducing inadvertent compensative thruster firings, and thus another source of inefficient propellant usage (inadvertent safe-mode entry triggering is another possible outcome).

Given all of this information, several questions beckon to be asked, namely:

1. Is there a way to prevent the drag-pass data gap?
2. Is there a way to improve our atmosphere models (reducing the 35% 1-sigma uncertainty)?
3. Is there a way to increase our knowledge of the spacecraft state?

The previous caveats and questions can be mitigated and answered with the navigation usage of Inertial Measurement Units (IMU), comprised of gyroscopes and accelerometers. The gyroscopes provide data pertaining to the spacecraft’s attitude/rotation, while the accelerometers provide data pertaining to the spacecraft’s translation. IMUs are especially sensitive to non-gravitational forces (i.e. precisely the environment not captured by current tracking techniques). This research demonstrates how to exploit this sensitivity for navigation performance, and thus reduce costs and risks.

Historically, IMUs have been used in space missions to quantify:

1. Maneuvers (i.e. Trajectory Correction Maneuvers {TCMs})
2. Orbit insertions (Lunar, Martian, etc.)
3. Atmospheric entries (i.e. Entry-Descent and Landing {EDL} operations, Shuttle re-entry)
4. Non-gravitational accelerations in order to refine Earth's gravity field (GRACE mission)

Thus far the IMU's use has been limited to taking the place of:

1. A thruster cutoff sensor (true for TCMs and aerobraking maneuvers (ABMs)).
2. Non-gravitational dynamic equations (true for Viking and Pathfinder landers).

These approaches are deterministic in that they do not account for the underlying statistics and cannot assist in reducing the uncertainties of the vehicle's state during aerobraking. Therefore, this research seeks to quantify statistical and actual improvements in spacecraft trajectory estimation by adding IMU data to radiometric-based orbit estimates. What this truly implies is that this approach provides for a completely independent way of measuring the spacecraft position and velocity during aerobraking that does not rely on DSN tracking. It also leads into the direction of a "smart" spacecraft, enabling a future mission to perform onboard navigation.

In the past several years, there has been work in the area of EDL (but not aerobraking) seeking to incorporate IMU data as navigation measurements instead of the traditional use of the data directly into the computed dynamics. Bob Bishop & Olivier Dubois-Matra, at UT Austin, have been developing and testing different approaches for this. At the Jet Propulsion Laboratory (JPL), Mike Lisano & Geoff Wawryzniak were developing and testing methods of doing this on the Mars Exploration Rovers (MER) mission, currently at Mars. They were developing a software package called IPANEMA (Interim Planetary Atmosphere Navigation for Estimation and Mission Analysis) for the 6-DOF reconstruction of the EDL trajectory of MER. Funding problems prevented its full development and use on MER. This aerobraking work takes advantage of IPANEMA's core capabilities. Previous work by the author in the area of using IMU data for aerobraking, funded through an autonomous aerobraking technology effort at JPL, revealed encouraging results. The current work builds upon those results.

2. METHODS AND PROCEDURES

In order to validate this data type for aerobraking navigation, two research routes can be taken: simulate and process the IMU data or process real IMU data. The benefit of simulating the data would be that one knows exactly what is in the data. However, this method can only theoretically validate the IMU data as a navigation metric. Processing real data is the preferred approach. The caveat to this is that it is usually more difficult (or time consuming) to achieve the results desired since reality is not necessarily ideal. The IMU data for the Mars Odyssey aerobraking phase was successfully collected and

archived. This research makes use of this data set in order to validate the use and implementation of IMU data for navigation. The following figure illustrates the normal aerobraking attitude for Mars Odyssey:

AEROBRAKING CONFIG - NORMAL

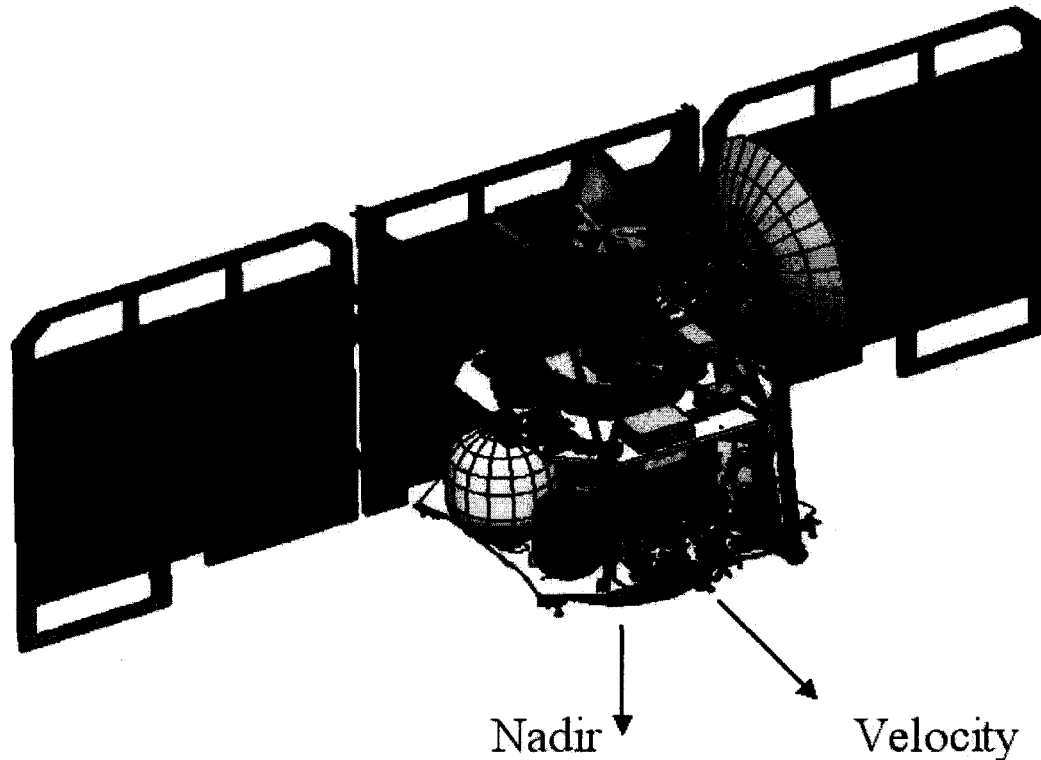


Figure 3. Mars Odyssey Normal Aerobraking Configuration. Spacecraft Axes: +Z (radial), +X (crosstrack), +Y (anti-intrack or anti-velocity).

In order to implement a data type for orbit determination, several questions need to be answered: What is the information content of the measurement? What needs to be done to the measurement in order for it to be useful? Does a relationship between what is being measured and what we want to know (estimate) exist and if so can it be formulated mathematically?

The acceleration measured by the IMU can be thought of as being comprised by the following elements:

1. Aerodynamic forces
2. Bias
3. Vibration
4. Angular motion
5. Solar Radiation Pressure

6. Thrusting (orbit or attitude maneuvers)

Given the previous list of non-conservative/rotational quantities, of particular interest are the effects of aerodynamic forces upon the spacecraft. The bias can easily be removed from the data because the IMU is operational *a priori* and *a posteriori* to the drag-pass. Any non-zero output in this regime is a bias since the non-conservative forces outside of the atmosphere (i.e. solar radiation pressure) are below the measurement threshold of the IMU. The spacecraft vibration can be removed, if some spectral analysis is performed and correlated with pre-launch “shake-tests”. The angular motion is also of interest in order to achieve a 6-DOF trajectory reconstruction. The angular motion will also induce a linear acceleration, sensed by the accelerometers due to their placement away from the spacecraft center-of-mass (COM). Thrusting events do not occur during the drag-pass, and therefore are not taken into account.

IPANEMA was created and originally intended for use in the EDL phase of the MER project. Due to funding issues, this software remained unfinished and in hibernation. Because IPANEMA’s structure lends itself for 6-DOF trajectory reconstruction of an atmospheric trajectory, its applicability to aerobraking is merely a matter of tailoring the modeled environment.

IPANEMA is Java based code that processes IMU (and altimeter if available) data in an orbit determination sense. IPANEMA was designed to be capable of estimating the translational and rotational states of a vehicle. Furthermore, knowing that the 6-DOF trajectory reconstruction will:

- a) be based on measurements that are noisy and potentially biased, and
- b) use some given combination of kinematics and user defined dynamics,

the total state vector formulation is partitioned in the following way:

$$\bar{\mathbf{X}}_T = \begin{Bmatrix} \mathbf{X}_k \\ \mathbf{X}_{pm} \\ \mathbf{X}_{pd} \end{Bmatrix} = \begin{Bmatrix} \text{19-element 6DOF kinematic parameter sub-vector} \\ \text{N}_m\text{-element measurement model parameter sub-vector} \\ \text{N}_d\text{-element dynamics model parameter sub-vector} \end{Bmatrix} \quad (1)$$

The IPANEMA core kinematic state vector X_k is defined as:

$$X_{k(19 \times 1)} = \begin{Bmatrix} \begin{bmatrix} \bar{r}_{cg} \end{bmatrix} & (3 \times 1) \\ \begin{bmatrix} \bar{v}_{cg} \end{bmatrix} & (3 \times 1) \\ \begin{bmatrix} \delta \bar{a}_{cg, unmod} \end{bmatrix}^B & (3 \times 1) \\ \begin{bmatrix} \bar{q}_{sc} \end{bmatrix}_B & (4 \times 1) \\ \begin{bmatrix} \bar{\omega}_{cg} \end{bmatrix}^B & (3 \times 1) \\ \begin{bmatrix} \delta \bar{\alpha}_{cg, unmod} \end{bmatrix}^B & (3 \times 1) \end{Bmatrix} = \left\{ \begin{array}{l} \text{vehicle c.g. position, meters, planet-ctrd inertial frame} \\ \text{vehicle c.g. velocity, m/sec, planet-ctrd inertial frame} \\ \text{vehicle c.g. unmodeled accelerations, m/sec}^2, \text{ vehicle} \\ \quad \text{c.g.-origin body frame} \\ \text{vehicle attitude quaternion, body frame-to-inertial frame} \\ \text{vehicle angular velocity, rad/sec, vehicle} \\ \quad \text{c.g.-origin body frame} \\ \text{vehicle unmodeled angular accelerations, rad/sec}^2, \\ \quad \text{c.g.-origin body frame} \end{array} \right\} \quad (2)$$

Note: cg = “center of gravity”; sc = “spacecraft”

X_{pm} and X_{pd} are user-defined/user-specific and are only processed if implemented by the user. Otherwise, IPANEMA only implements the core kinematic 19×1 state. The user also defines the force models, torque models, spacecraft physical parameters, and planetary model parameters of interest. IPANEMA can estimate the kinematic state of a vehicle that is a box being kicked down a hill, or a spacecraft going through an atmosphere. IPANEMA is generic in that sense. It is the specific user that tailors the way that IPANEMA is implemented. This is where IMAN comes into existence. IMAN is Inertial Measurements for Aerobraking Navigation (IMAN), and it is a Java/Python-based user-specific implementation of IPANEMA.

In computer science/engineering vernacular, IMAN is a set of user-defined classes that extend abstract classes that make up IPANEMA. The class “ImanDynamics” contains methods related to propagating the spacecraft state and associated covariance. “ImanKineticModels” is a class that contains methods related to common force/torque models to be implemented in the aerobraking dynamic and observation models. “ImanObservations” is a class that provides methods of computing partial derivatives and residuals of IMU observations. “ImanMain” is the Python interface where the initial conditions (state, covariance), measurement noise, process noise parameters, etc. are defined.

The following figure illustrates the various components of IPANEMA and IMAN, and how they interface. Everything to the right of the horizontal dashed line are the user-defined IMAN classes; IPANEMA is to the left:

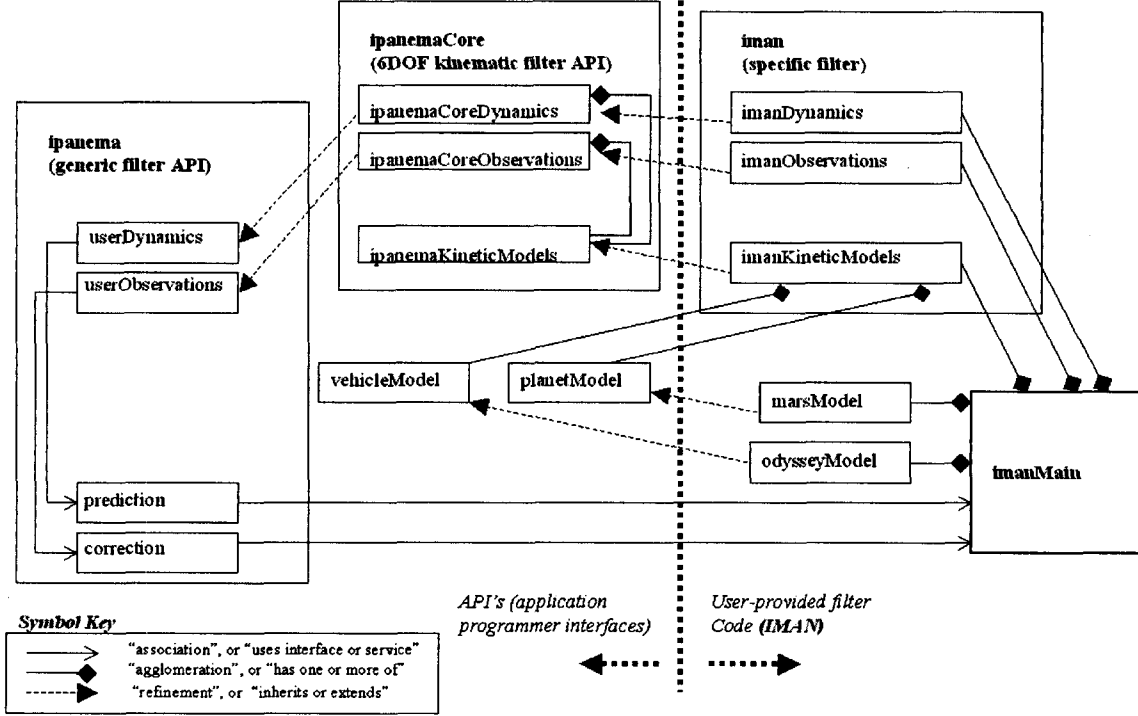


Figure 4. IPANEMA/IMAN interface.

The data set obtained for Odyssey (200Hz), already has the biases removed from the IMU and the angular-motion-induced linear accelerations removed from the accelerometers. The data is in the spacecraft body frame, and thus needs not be transformed from the IMU case frame to the body frame. However, if this were not the case, biases would need to be estimated and removed and the angular motion signature in the accelerometers would also need to be removed. The gyroscope and accelerometer measurement equations are as follows:

$$\bar{\omega}_{G(3 \times 1)} = M_{SFg(3 \times 3)} \left\{ I_{(3 \times 3)} + M_{MAg(3 \times 3)} + M_{NOg(3 \times 3)} \right\} \left\{ T_{B(3 \times 3)}^I \left[\bar{\omega}_{sc(3 \times 1)} \right]^B \right\} + \bar{b}_{g(3 \times 1)} + \bar{e}_{g(3 \times 1)} \quad (3)$$

Where:

- $\bar{\omega}_G$ = Angular rates measured by gyro, in gyro platform frame
- M_{SFg} = 3×3 matrix of gyro scale factor errors (diagonal matrix)
- M_{MAg} = 3×3 matrix of gyro axis misalignment errors (zero-diagonal matrix)
- M_{NOg} = 3×3 matrix of gyro axis non-orthogonality errors (zero-diagonal matrix)
- \bar{b}_g = 3×1 vector of gyro biases, per axis

$\bar{\epsilon}_g$ = 3×1 vector of random gyro noise

For a gyroscope reporting unbiased rates, with no scale factor errors in unit conversion, no misalignment with the nominal gyro platform frame, nor axis-to-axis nonorthogonality, the measurement model is

$$\bar{\omega}_G = T_B^G [\bar{\omega}_{sc}]^B \quad (4)$$

Equation 4 is the default gyro measurement modeled in the observations class of IPANEMA Core Kinematics. Note that $[\bar{\omega}_{sc}]^B$ is part of core kinematic state vector X_k . Other correction terms, such as biases, scale factors, etc., are supplied by the user, in any fashion the user deems. Again, LMA removes these quantities from the data. Define T_a^B as a 3×3 matrix transforming the three accel axes to the vehicle frame. If $[\bar{a}_a]^B$ is the actual specific force expressed in vehicle body frame coordinates, then accelerometer output measurements are given by:

$$\bar{a}_a = M_{SFa}_{(3 \times 3)} \left\{ I_{(3 \times 3)} + M_{MAa}_{(3 \times 3)} + M_{NOa}_{(3 \times 3)} \right\} \left\{ T_{B(3 \times 3)}^a [\bar{a}_{a(3 \times 1)}]^B \right\} + \bar{b}_{a(3 \times 1)} + \bar{\epsilon}_{a(3 \times 1)} \quad (5)$$

Where:

\bar{a}_a = Specific force measured by accelerometer “triad,” in accel platform frame
 M_{SFa} = 3×3 matrix of accel scale factor errors (diagonal matrix)
 M_{MAa} = 3×3 matrix of accel axis misalignment errors (zero-diagonal matrix)
 M_{NOa} = 3×3 matrix of accel axis non-orthogonality errors (zero-diagonal matrix)
 \bar{b}_a = 3×1 vector of accel biases, per axis
 $\bar{\epsilon}_a$ = 3×1 vector of random accel noise

For an accelerometer reporting unbiased specific forces, with no scale factor errors in unit conversion, no misalignment with the nominal accelerometer platform frame, nor axis-to-axis non-orthogonality, the measurement model is

$$\bar{a}_a = T_B^a [\bar{a}_a]^B \quad (6)$$

Where:

$$\begin{aligned}
\left[\bar{a}_{a(3 \times 1)} \right]^B &= \left\{ T_{I(3 \times 3)}^B \left[\bar{a}_{cg, ng, mod(3 \times 1)} \right]^I \right\} + \left[\tilde{\alpha}_{cg, unmod(3 \times 1)} \right]^B \\
&+ \left\{ \left[\bar{\omega}_{sc(3 \times 1)} \right]^B \times \left[\bar{\omega}_{sc(3 \times 1)} \right]^B \times \left[\bar{r}_{a/cg(3 \times 1)} \right]^B \right\} \\
&\left\{ \left\{ \left[\bar{\alpha}_{sc, mod(3 \times 1)} \right]^B + \left[\bar{\alpha}_{sc, unmod(3 \times 1)} \right]^B \right\} \times \left[\bar{r}_{a/cg(3 \times 1)} \right]^B \right\}
\end{aligned} \tag{7}$$

In Eq. 7, the terms on the right-hand side are:

- T_I^B = Transformation matrix, inertial frame to body frame
- $\left[\bar{a}_{cg, ng, mod} \right]^I$ = Acceleration of the vehicle c.g. due to modeled, non-gravitational forces acting on the vehicle (inertial frame)
- $\left[\tilde{\alpha}_{cg, unmod} \right]^B$ = Unmodeled acceleration of the vehicle c.g.; part of core kinematic state vector (vehicle body frame)
- $\left[\bar{\omega}_{sc} \right]^B$ = Angular velocity of the vehicle; part of core kinematic state vector (vehicle body frame)
- $\left[\bar{r}_{a/cg} \right]^B$ = Distance vector from vehicle c.g., to accelerometer “triad origin” reference point (vehicle body frame)
- $\left[\bar{\alpha}_{sc, mod} \right]^B$ = Angular acceleration of the vehicle due to modeled torques (including gravitational) (vehicle body frame)
- $\left[\bar{\alpha}_{sc, unmod} \right]^B$ = Unmodeled angular acceleration of the vehicle; part of core kinematic state vector (vehicle body frame)

Looking more closely at two of the above terms, we see that modeled forces and torques, which also appear in IPANEMA Core Kinematics equations of motion, play a role in modeling the accelerometer measurement:

$$\left[\bar{a}_{cg, ng, mod} \right]^I = \frac{1}{\langle M_{sc} \rangle^A} \left\langle \left[\bar{F}_{ng, dyn} \right]^I \right\rangle^A \tag{8}$$

$$\left[\bar{a}_{sc, mod} \right]^B = \left\{ \left\langle I_{sc} \right\rangle^A \right\}^1 \left\{ T_I^B \left\langle \left[\bar{\tau}_{dyn} \right]^I \right\rangle^A - \left[\bar{\omega}_{sc} \right]^B \times \left\langle I_{sc} \right\rangle^A \left[\bar{\omega}_{sc} \right]^B \right\} \tag{9}$$

Where M_{sc} , $[\bar{F}_{ng, dyn}]^T$, I_{sc} , and $[\bar{\tau}_{dyn}]^T$ are user-provided, abstract values for modeled vehicle mass, nongravitational force, vehicle inertia tensor, and torques.

In order to process the IMU data, initial conditions must first be obtained. Since real data is being used, the initial state and covariance are taken from the Odyssey navigation solution. Radiometric data is fit from the previous drag-pass up to the turn-to-drag-pass attitude for the drag-pass of interest. Once the radiometric data is fit, a state and covariance are mapped to the time of the IMU initial observations. This state and covariance are read into IMAN as its initial conditions.

The spacecraft mass is the same assumed by the navigation team, for a given drag-pass. The normal projected area for each spacecraft axis was also taken from the navigation team's assumptions. A variable effective area model, dependent on the spacecraft attitude, is implemented in IMAN. Odyssey navigation used a variable drag coefficient model that was altitude dependent. This is not implemented in IMAN. Odyssey navigation also used the Mars Global Reference Atmospheric Model (MarsGRAM), by Gere Justus.

This is not the atmosphere model implemented in IMAN. However, MarsGRAM is used in order to generate parameters for an exponential model. The exponential model used was obtained from Dr. Bob Bishop at the University of Texas at Austin. A data fit was made to Viking data, and the parameters from that fit were initially used for the IMAN atmosphere model. The following equation describes the exponential atmosphere used.

$$\rho = \rho_o e^{-C_1 \frac{h}{H_o} + C_2 \cos\left(\frac{2\pi h}{H_o}\right) + C_3 \sin\left(\frac{2\pi h}{H_o}\right)} \quad (10)$$

The traditional exponential density model has $C_1 = 1$ and $C_2 = C_3 = 0$, with the scale height (H_0) and the base density (ρ_o) consistent with the local layer of atmosphere. The manner in which MarsGRAM is employed and the exponential atmosphere generated, is as follows. The predicted periapse latitude, longitude, and time, are passed into MarsGRAM. MarsGRAM is setup to output density and density uncertainty at 1 kilometer altitude increments from about 90 kilometers to 500 kilometers. The MarsGRAM density/altitude outputs are read into a program developed by Mike Lisano in order to do the curve fit. 5 parameters are output from this code: ρ_o , H_0 , C_1 , C_2 , and C_3 (with their associated covariance). These are then input into IMAN.

3. RESULTS

The first implementation of IMAN consists of solely estimating for the core kinematic state parameters (i.e. no observation or dynamic model states). 200Hz data was taken and processed. The filter used is a Batch - Extended Kalman Filter (EKF). This means that at any given measurement time, there are 6 measurements to process (3 accelerometers and 3 gyroscopes). At each time, the 6 measurements are accumulated and a batch-least-squares fit is performed. The state is then updated with the current best state estimate. Process noise is also used in this filter. There are 3 unmodeled acceleration states and 3 unmodeled angular acceleration states.

The Viking generated atmosphere model was far from reflecting the observed values of acceleration. It was insensitive to the observed lift and sideforce due to the fact that the attitude had not been modeled taking into account the information provided to us by the gyroscopes. At first, the attitude was modeled by assuming a pitch rate consistent with the time rate of change of the true anomaly.

The next two steps consisted of refining the atmosphere model in order to get values that were more representative of the actual drag pass, and to incorporate the knowledge of the attitude provided by the gyroscopes. In order to achieve a more realistic attitude profile, the attitude is modeled by tuning the gyroscope measurement noise and the process noise uncertainties and time constants. Once realistic values for the measurement noise were obtained, the gyro fit was mostly sensitive to the process noise time constant then the associated uncertainty. Mike Lisano noticed a significant sensitivity to the base density. The two most sensitive atmospheric model parameters are the base density and the C1 term.

The following figures illustrate preliminary results obtained with the 200Hz and 1Hz smoothed data. Note that only 1 Hz results are shown for the gyro data since changes between the 200Hz and 1Hz results are less apparent for the gyros than for the accelerometers.

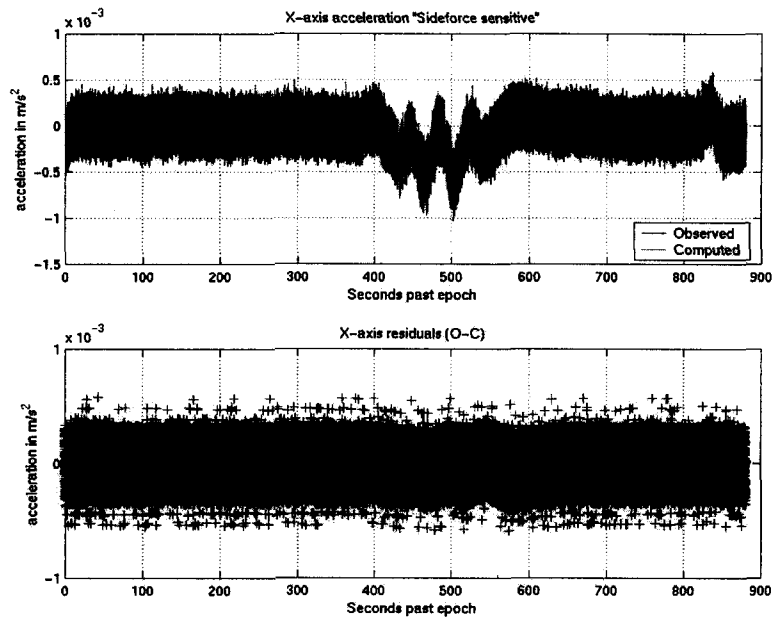


Figure 5. X-axis accelerometer data (200 Hz)

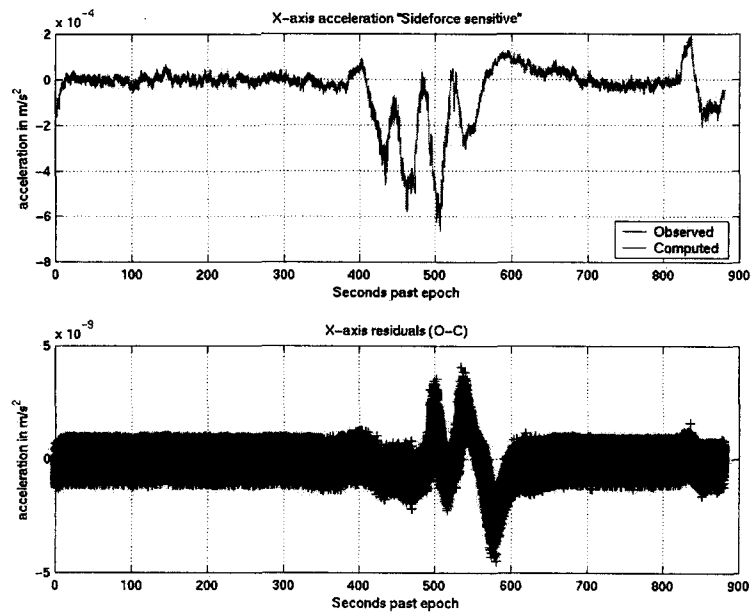


Figure 6. X-axis accelerometer data (1 Hz sliding average)

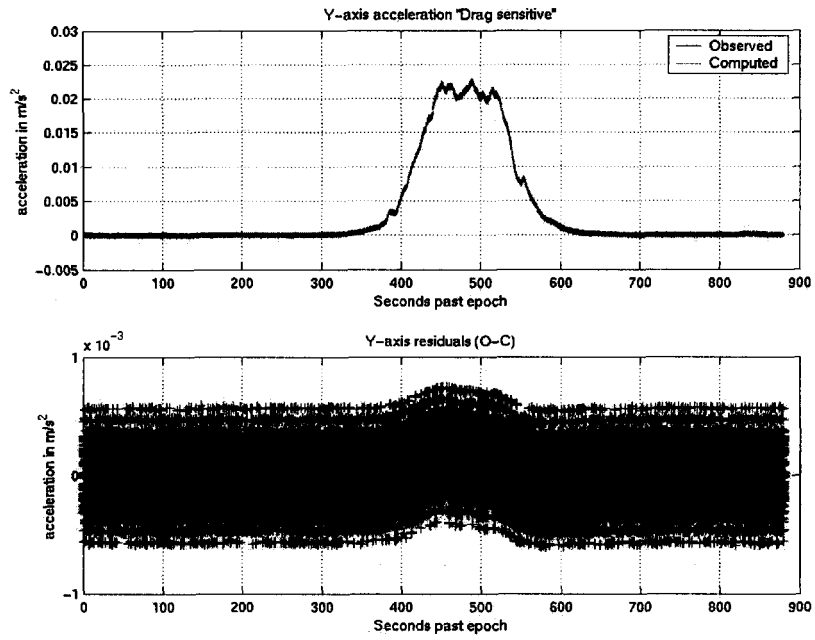


Figure 7. Y-axis accelerometer data (200 Hz)

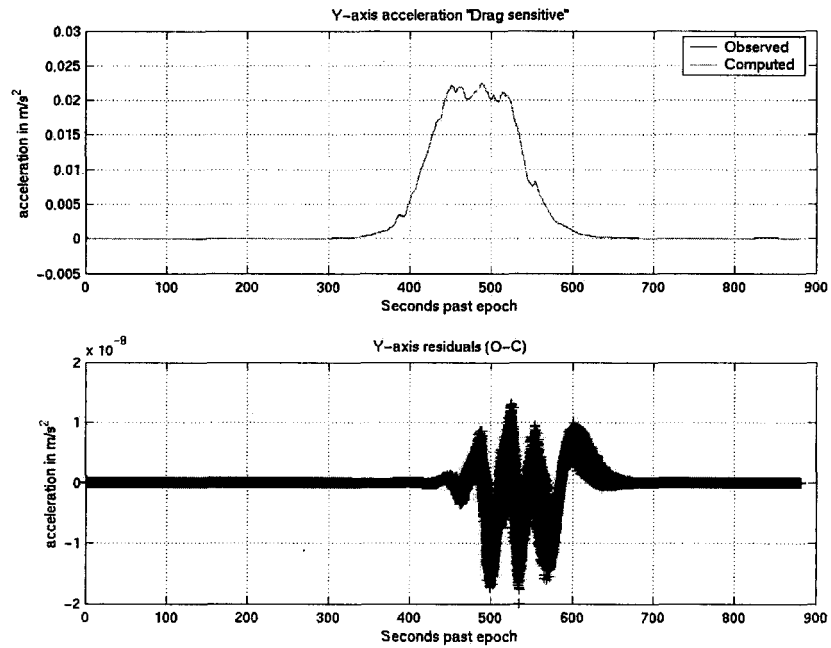


Figure 8. Y-axis accelerometer data (1 Hz sliding average)

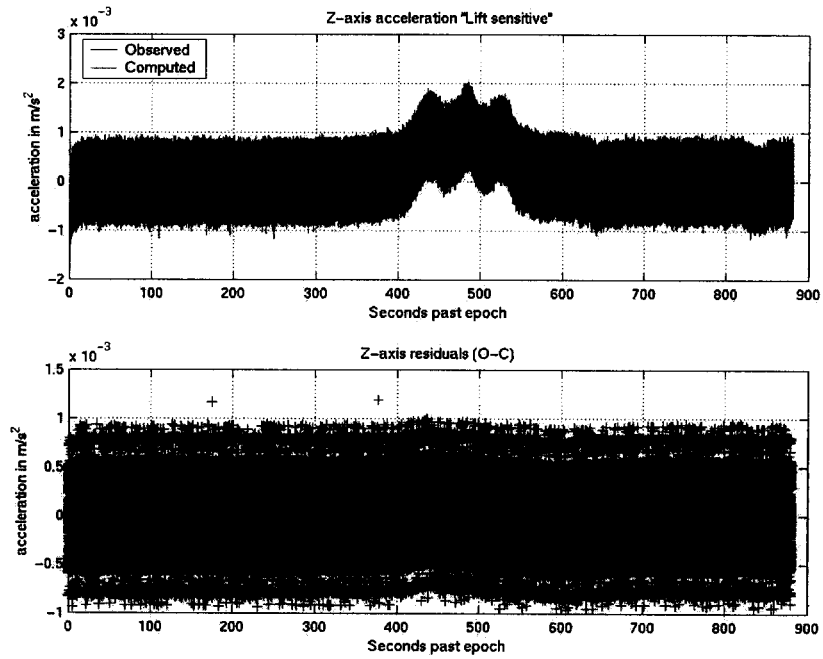


Figure 9. Z-axis accelerometer data (200 Hz)

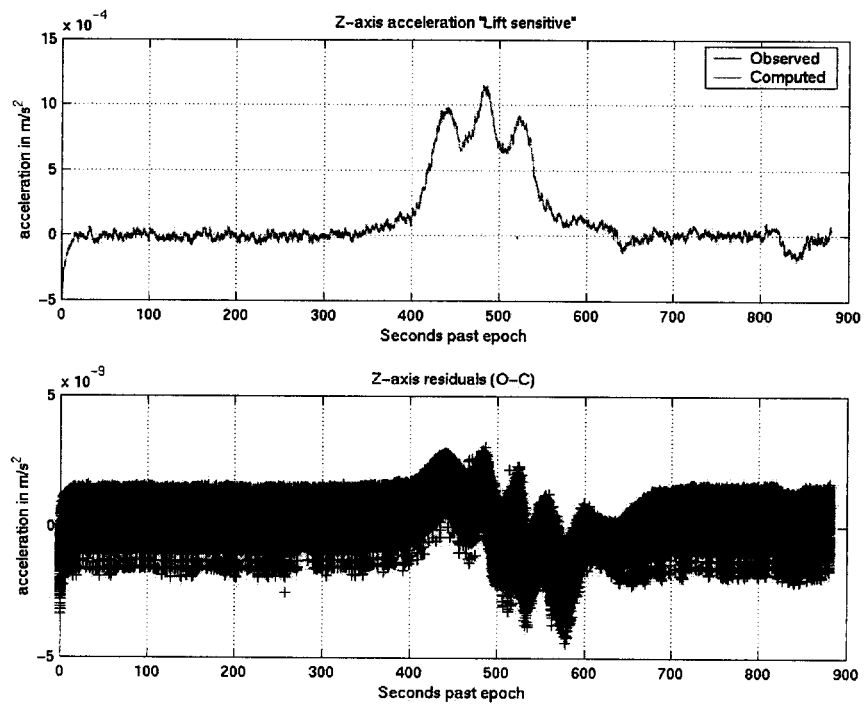


Figure 10. Z-axis accelerometer data (1 Hz sliding average)

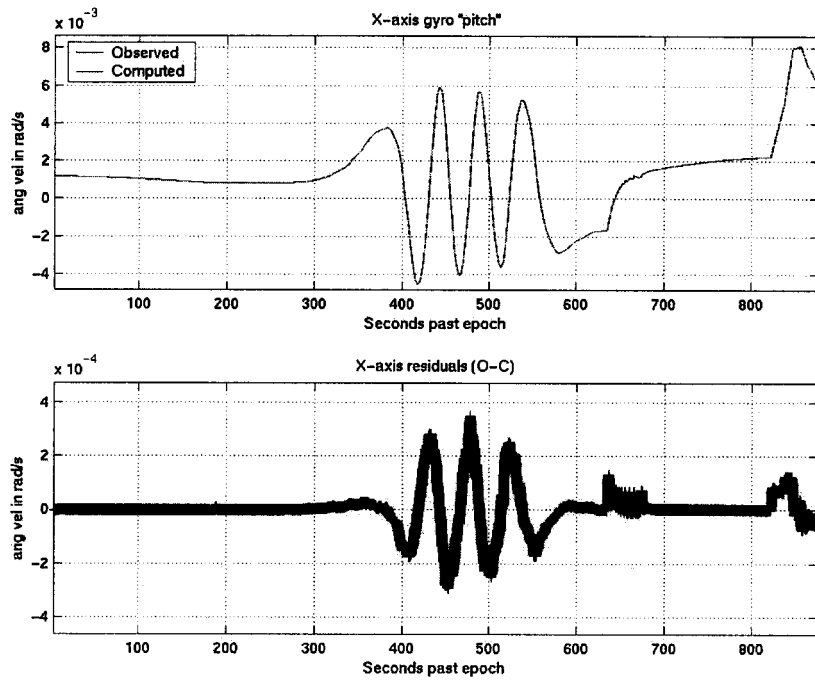


Figure 11. X-axis gyro data (1 Hz sliding average)

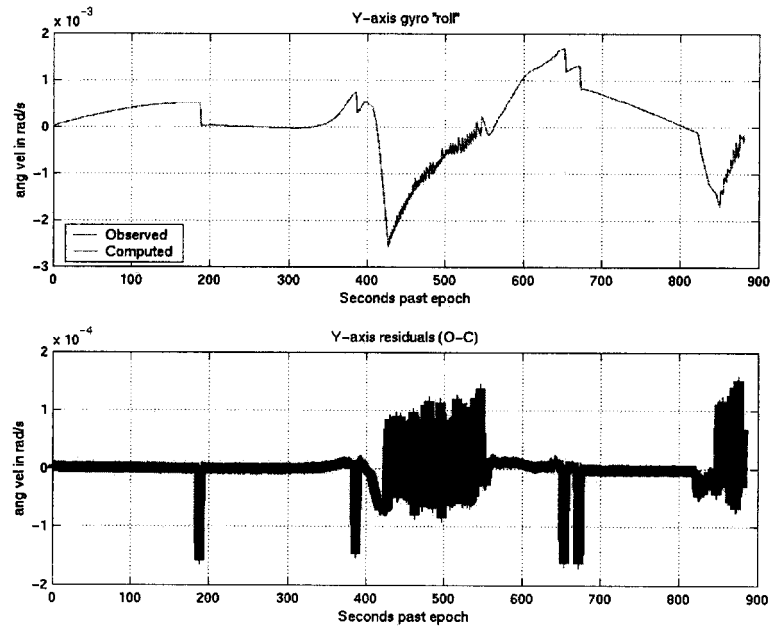


Figure 12. Y-axis gyro data (1 Hz sliding average)

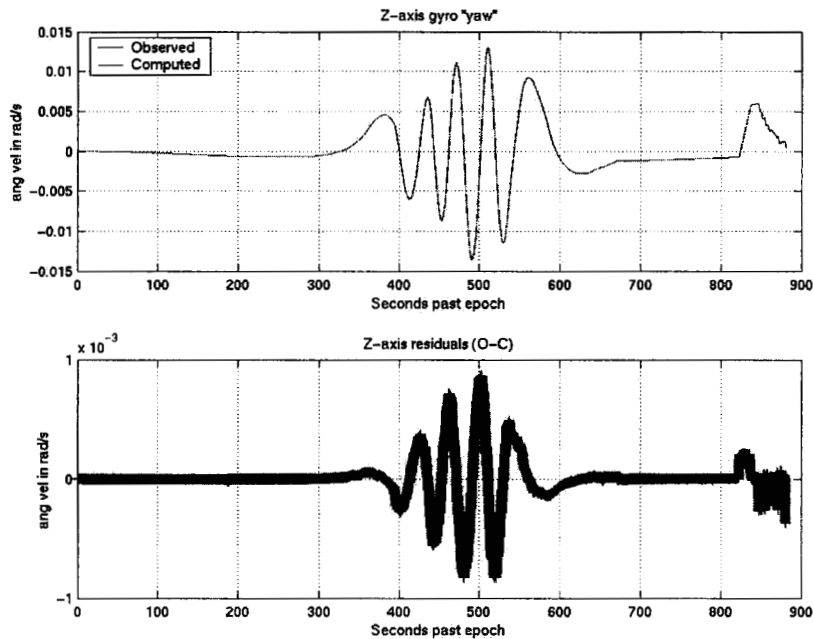


Figure 12. Z-axis gyro data (1 Hz sliding average)

4. CONCLUSIONS AND RECOMMENDATIONS

During aerobraking, telemetry is typically lost during the drag pass. Being that IMUs are a part of typical payloads, there may be advantages in using the IMU data to aid in aerobraking navigation capabilities. There are aspects of the aerobraking process that could benefit from being automated (and a must, for outer planets). Thus far, there have been positive indicators that IMU data may be useful in enhancing the spacecraft navigation scheme.

The results thus far show that it is possible to fit an aerobraking trajectory in a 6-DOF sense, by using IMU data as a navigation measurement. A pure kinematic state covariance shows realistic uncertainties in the state given some nominal atmospheric model. The natural steps to be performed in the process of implementing IMU as a data type are:

1. add in a higher fidelity gravity model (perhaps up to a 4x4 field)
2. estimate for observation model states, namely base density and scale height
3. continue the filter-parameter tuning process
4. compare the results against navigation team solutions
5. fit representative aerobraking orbits of various orbital periods (e.g. 16 hours, 8 hours, and 3 hours)

6. use various filtering strategies in order to determine what suits the IMU data the best (i.e. Unscented Kalman Filter, Square Root Information Filter, UDU Filter, etc.)

Using the raw 200 Hz data will not yield the results desired because of its undesirable noise level. The data must be preprocessed (smoothed to a lesser frequency) in order to drive the data noise down and allow for a greater probability of reducing the state uncertainties. Also, the IMU *a priori* covariance obtained from the radiometric data fit may be optimistic. If the true error is outside of the *a priori* covariance, then the problem may be over constrained and the results obtained may be unrealistic. Therefore, the IMU data sensitivity to the *a priori* covariance will be analyzed.

Solely using the real data makes it difficult to obtain the filter tuning parameters in a timely manner, and in fact they may never be obtained. Thus, IMU data will be simulated in an attempt to converge upon the proper filter tuning parameters in a more expeditious manner, and then applying those parameters to process the real data.

5. ACKNOWLEDGEMENTS

The work described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

The authors would like to gratefully acknowledge the help and contributions of the following people: Tung-Han You, Dolan Highsmith, Dan Johnston, Jet Propulsion Laboratory; George Born, Penina Axelrad, Colorado Center for Astrodynamics Research.

6. REFERENCES

1. Wawrzyniak, G., Lisano, M., “*Using Inertial Measurements for the Reconstruction of 6-DOF Entry, Descent, and Landing Trajectory and Attitude Profiles*”, AIAA/AAS Astrodynamics Specialist Conference, AAS 02-164, 2002.
2. Bishop, R., “*Precision Entry Navigation Dead-Reckoning Error Analysis – Theoretical Foundations*”, collection of personal notes, 2001.
3. Bishop, R., Dubois-Matra, O., “*Development of a Planetary Entry Model Suitable for Processing Acceleration Measurements*”, Interim Report #1 to Todd Ely (JPL), 2001.
4. Bishop, R., Dubois-Matra, O., “*Entry Navigation Extended Kalman Filter for IMU Processing*”, Interim Report #2 to Todd Ely (JPL), 2001.

5. Jah, M.K., “*A Proposed Use of Accelerometer Data for Autonomous Aerobraking at Mars*”, AIAA/AAS Astrodynamics Specialist Conference , AAS 01-386, 2001.
6. Lisano, M., “*Interim Planetary Atmosphere Navigation for Estimation and Mission Analysis (IPANEMA) Algorithm Design Document (ADD) Version 1.0*”, JPL document, 2002.
7. Mase, R., Antreasian, P., Bell, J., Martin-Mur, T., “*The Mars Odyssey Navigation Experience*”, AIAA/AAS Astrodynamics Specialist Conference, AIAA 2002-4530, 2002.
8. Esposito, P., Alwar, V., Demcak, S., Graat, E., Johnston, M., Mase, R.A., “*Mars Global Surveyor: Navigation and Aerobraking at Mars*”, AIAA/AAS Astrodynamics Specialist Conference AAS 98-384, 1998.